ATTORNEY'S DOCKET NO: **99-TK-551SS**

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant:	David Alan Edwards, et al.
Serial No:	09/411,792
Confirmation No:	8808
Filed:	October 1, 1999
For:	INTERFACE FOR TRANSFERRING DEBUG INFORMATION
Examiner:	Ted T. Vo
Art Unit:	2122

Certificate of Electronic Filing Under 37 CFR 1.8
I hereby certify that this paper (along with any paper referred to as being attached or enclosed) is being transmitted via the Office electronic filing system in accordance with § 1.6(a)(4).

Dated: 6-10-08

Eileen M. MacKenzie

Board of Patent Appeals and Interferences
United States Patent and Trademark Office
P. O. Box 1450
Alexandria, VA 22313-1450

APPELLANT'S APPEAL BRIEF PURSUANT TO 37 C.F.R. §41.37

Scott J. Gerwin, Reg. No. 57,866
James H. Morris, Reg. No. 34,681
Wolf, Greenfield & Sacks, P.C.
600 Atlantic Avenue
Boston, Massachusetts 02210
(617) 720-3500
Attorneys for Appellants

TABLE OF CONTENTS

This brief is in furtherance of the Notice of Appeal mailed on April 10, 2008. In accordance with 37 C.F.R. §1.8(a), the deadline for submission of this appeal brief under 37 C.F.R. §41.37(a)(1) is June 10, 2008. The fee required under 37 C.F.R. §41.20(b)(2) is submitted herewith.

I. REAL PARTY IN INTEREST (37 C.F.R. §41.37(c)(1)(i))

The real party in interest in this application is the assignee, ST Microelectronics Limited, a corporation having a place of business at 1000 Aztec West, Almondsbury, Bristol BS12 4SQ, United Kingdom.

II. RELATED APPEALS AND INTERFERENCES (37 C.F.R. §41.37(c)(1)(ii))

There are no other appeals or interferences known to the Appellant, the Appellant's legal representative, or the assignee which will directly affect, be directly affected by, or have a bearing on the Board's decision in this appeal.

III. STATUS OF CLAIMS (37 C.F.R. §41.37(c)(1)(iii))

There are 64 total claims currently pending in this application (i.e., claims 1-64), of which four are independent, and sixty are dependent. Each of claims 1-64 stands rejected and each of these claims is appealed. The appealed claims are set forth in Appendix A. The status of each of the claims is summarized in the list below:

1. Rejected and Appealed: Claims 1-64
2. Allowed: None
3. Withdrawn: None
4: Objected To: None
5. Canceled: None

IV. STATUS OF AMENDMENTS (37 C.F.R. §41.37(c)(1)(iv))

No amendments have been made subsequent to the mailing of the Final Office Action of January 10, 2008.

V. SUMMARY OF CLAIMED SUBJECT MATTER (37 C.F.R.§41.37(c)(1)(v))

In one aspect, at least one embodiment of the invention relates generally to an interface for transferring debug information for debugging a processor. (Specification, page 1, lines 6-7). System-on-chip devices are devices that include a processor, one or more modules, bus interfaces, memory devices, and one or more system buses for communicating information on a single chip (Specification, page 1, lines 10-12). Because communications between modules occur internally to the chip, accessing this information to diagnose hardware or software problems presents challenges. (Specification, page 1, lines 12-15).

A non-limiting example of one embodiment described in Appellant's specification is provided below. As shown in Figure 1, integrated circuit 101 includes a processor 102 and a debug circuit 103, coupled by a system bus 105 (Specification, page 7, lines 2-5) System bus 105 may be, for example, a conventional processor bus, packet switch, or other communication medium used to communicate operating information between modules of device 101 (Specification, page 7, lines 5-7). Communication link 104 couples processor 102 to debug circuit 103, and is separate from system bus 105 (Specification, page 7, lines 20-21). Communication link 104 is configured to transfer debug information from processor 102 to debug circuit 103, and to transfer state and processor control information from the debug circuit 103 to processor 102 (Specification, page 7, lines 23-25). The information sent from the processor to the debug circuit may include, for example: operand address information; (Specification, page 10, lines 6-10); an operand value (Specification, page 7, lines 23-25); a program counter value (Specification, page 8, lines 30-33); a status indicating that a computer instruction in the writeback stage is a valid instruction (Specification, page 9, lines 4-5); a status indicating that a computer instruction in the writeback stage is a first instruction past an executed branch instruction (Specification, page 9, lines 6-8); a status indicating a type of the executed branch instruction (Specification, page 9, lines 14-16); and/or process identifier information of an executed instruction (Specification, page 9, line 31 – page 10, line 3).

A discussion of each independent claim is provided below, with reference to the specification by page and line number. The portions of the specification referred to below describe examples of embodiments of the invention and the claims are not limited to these particular examples.

Claim 1

Claim 1 is directed to a microcomputer comprising a processor and a debug circuit. The processor and the debug circuit may be coupled by a communication link, which is used by the processor to transmit information about the state of the processor, including at least an operand address that indicates a memory location at which an operand value is stored (Specification, page 7, lines 2-5; page 7, lines 20-25).

Claim 21

Claim 21 is directed to a microcomputer on a single integrated circuit, comprising a processor and a debug circuit (Specification, page 7, lines 2-5). The processor and the debug circuit may be coupled by a communication link, which is used by the processor to transmit information about the state of the processor, including at least one of: an operand address that indicates a memory location at which an operand value is stored and an operand value (Specification, page 7, lines 20-25). The processor is further configured to transmit to the debug circuit a program counter value, a status indicating that a computer instruction in the writeback stage is a valid instruction, a status indicating that a computer instruction in the writeback stage is a first instruction past an executed branch instruction, a status indicating a type of the executed branch instruction, and process identifier information of an executed instruction (Specification, page 8, lines 30-33; page 9, lines 4-16; page 9, line 31— page 10, line 3).

Claim 22

Claim 22 is directed to a microcomputer comprising: at least one processor (Specification, page 7, lines 2-5); a debug circuit (Specification, page 7, lines 2-5); a system bus

(Specification, page 7, lines 2-5); and means for transmitting to the debug circuit a plurality of bit values each representing a state of an operation in the processor including at least an operand address that indicates a memory location at which an operand value is stored (Specification, page 7, lines 20-25).

The limitation of claim 22 reciting a "means for transmitting" is a means-plus-function limitation and its corresponding structure is described in numerous locations in the specification, including, for example, page 7, lines 20-26 and the description accompanying element 104 of Figure 1; page 8, lines 24-29 and the description accompanying element 215 of Figure 2; page 12, lines 1-8 and the description accompanying element 313 of Figure 3; page 14, lines 4-5, and the description accompanying element 420 of Figure 4.

Claim 42

Claim 42 is directed to a method of transferring information between a processor and a debug circuit over a communication link (Specification, page 7, lines 2-5). The method comprises: transmitting to the debug circuit a plurality of bit values each representing a state of an operation in the processor including at least one operand address that indicates a memory location at which an operand value is stored (Specification, page 7, lines 20-25); and transmitting a program counter value indicating the program counter of the processor (Specification, page 8, lines 30-33).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL (37 C.F.R.§41.37(c)(1)(vi))

The sole ground of rejection to be reviewed on appeal is the rejection of claims 1-64 under 35 U.S.C. §102(b) as purportedly being unpatentable over U.S. Patent No. 5,737,516 (hereinafter "Circello") and "Debug Support on the ColdFire Architecure," Motorola, Inc.

VII. ARGUMENT (37 C.F.R. §41.37(c)(1)(vii))

As discussed above, claims 1-64 stand rejected by the Office Action mailed January 10, 2008 under 35 U.S.C. §102(b) as purportedly being unpatentable over Circello and Debug Support on the ColdFire Architecture (referred to in the Office Action and hereinafter as "MotorolaNPL"). This rejection is improper because: (1) the Examiner has made an improper multiple reference rejection under §102; and (2) Circello and MotorolaNPL fail to disclose all of the limitations of the independent claims.

A. Discussion of Circello

Figure 1 of Circello shows a data processing system 5 that includes a processing core 9 and a debug module 10. As shown in Figure 1, several signals are transmitted from the processing core 9 to the debug module 10. A Bus Grant signal may be transmitted from processing core 9 to debug module 10 which is used by CPU 2 of processing core 9 to indicate to debug module 10 that it has been granted use of K-Bus 25 (Col. 29, lines 41-55). Further, a CPU Processor Status (CPST) signal may be transmitted from processing core 9 to debug module 10 that indicates the type of operation currently being executed by data processor 3 (Col. 18, lines 47-53). For example, the CPST signal may indicate when execution of an instruction begins, when execution of an instruction should continue, when data processor 3 enters into a selected mode of operation, when a preselected branch instruction is executed, and when operation of data processor 3 is halted (Col. 18, lines 53-60).

Additionally, several signals may be transmitted from processing core 9 to debug module 10 over K-Bus 25 (Col. 4, lines 62-63). The signals transmitted over K-Bus 25 are KADDR, KDATA, and KCONTROL (Col. 4, lines 63-65). The KADDR signal is used to send instruction addresses accessed during normal operation of data processor 3 to debug module 10 and the KDATA signal is used to send operand values to debug module 10 (col. 18, lines 25-31). Circello does not explicitly disclose how the KCONTROL signal is used.

In summary, in the system of Circello there are five signals transmitted from processing core 9 to debug module 10 (Bus Grant, CPST, KADDR, KDATA, and KCONTROL). Circello discloses that these signals are used to transmit information such as CPU status, instruction addresses, and operand values.

Nowhere does Circello disclose or suggest that operand addresses are sent to debug module 10 via these signals. Further, Circello does not disclose or suggest that the processing core 9, debug module 10, and K-bus 25 are all implemented on a single integrated circuit.

B. Discussion of MotorolaNPL

MotorolaNPL is a publication that describes the same system that is described in Circello. Indeed, the inventors of Circello, Joseph Circello and William Hohl, are listed as authors of MotorolaNPL, and the Examiner appears to have acknowledged the relationship between these two publications as, in the Final Office Action mailed January 10, 2008, he states, "[f]urthermore, it should be noted that Circello in the Motorola Patent and Motorola's NPL is the same person."

Because these two references describe the same system, MotorolaNPL does not add any disclosure that is relevant to Appellant's claims beyond what is disclosed by Circello. As shown in Figure 3 on page 2 of MotorolaNPL, a debug module is coupled to a processor core and on-chip memory via a K-Bus.

Figure 4 on page 3 of Motorola NPL shows a more detailed block diagram of the debug module. MotorlaNPL discloses that the debug module has a parallel output port that delivers encoded processor status and data. Two signals, PST and DDATA, are output via this port. The

PST signal indicates the execution status of the core and the DDATA signal outputs operand data.

Figure 5 on page 4 of MotorolaNPL shows the signals that can be written from the CPU core to the debug module. These signals are clk, kaddr, kdata, kta, cpu_wrt_drc, and rdata. Nowhere does MotorolaNPL disclose or suggest that any of these signals transmit operand address information. Rather, the clk signal is a clock signal, the kta signal is used as a transfer acknowledgment signal, the cpu_wrt_drc signal is used by the CPU core to inform the CPU that a debug instruction is being loaded into the RDATA register, and the rdata signal appears to be used to load instructions into the RDATA register.

MotorolaNPL is silent as to what information is transmitted by the kaddr and kdata signals, but it appears that these signals are the same KADDR and KDATA signals discussed in Circello, which are used to transmit instruction addresses and operand values, respectively, from the CPU core to the debug module.

Nowhere does MotorolaNPL disclose or suggest that operand addresses are sent to the debug module via these signals. Further, MotorlaNPL does not disclose or suggest that the CPU core and debug module are implemented on a single integrated circuit.

C. Neither Circello Nor Motorola NPL Discloses Transmitting An Operand Address From A Processor To A Debug Circuit

During the lengthy prosecution of this application, Appellant has repeatedly pointed out that neither Circello nor MotorolaNPL discloses transmitting an operand address from the processor core to the debug circuit.

Appellant emphasizes that neither Circello nor MotorolaNPL discloses or suggests that any of the signals sent from the processor core to the debug module include an operand address. Although the KADDR signal may include instruction addresses, instruction addresses are very different from operand addresses. That is, an instruction address specifies the memory location of an instruction stored in memory, whereas an operand address specifies the memory location of an operand value stored in memory.

The Final Office Action mailed January 10, 2008 appears to assert that Circello and Motorola disclose transmitting an operand address from the processor to a debug circuit in a number of places. In particular, the Office Action asserts that Circello discloses this aspect of the invention at col. 20, lines 27-61, col. 14, lines 65-67, and col. 15, lines 1-23. *See* Final Office Action of January 10, page 4. The Office Action also asserts that MotorolaNPL discloses this aspect of the invention in section 3 on page 4 and in section 4 on pages 5-6. *See* Final Office Action, pages 2 and 4.

However, none of these cited portions of the references discloses or suggests transmitting an operand address from the processor to a debug circuit. Indeed, none of these cited portions even mentions the term "operand address." At col. 20, lines 27-61 of Circello, Circello discusses capturing an instruction address. An instruction address is very different from an operand address, as an instruction address is a memory address at which an instruction is stored, whereas an operand address is a memory address at which an operand is stored.

At col. 14, lines 65-67, Circello does not mention operand addresses, but rather states that, "the present invention provides an important and hereto nonexistent, real time trace function." This portion of Circello does not say anything about operand addresses or transmitting an operand address from a processor to a debug circuit. Col. 15, lines 1-23 of Circello discusses the DDATA and PST signals that are output by the debug module, but does not disclose or suggest that these signals provide operand address information. Quite to the contrary, this portion of Circello states, "[t]he DDATA signal provides data which reflects operand data and that PST signals provide encoded status information which reflects an execution status of CPU 2. Furthermore, the DDATA signal also provides captured instruction address program flow changes to allow an external development system to trace an exact program flow without requiring externally visible address bus or an externally visible data bus."

Thus, Circello discloses that the DDATA and PST signals output by the debug circuit provide instruction address information, operand value information, and processor execution status information, but does not disclose or suggest that these signals provide operand address information.

The cited portions of MotorolaNPL also fail to disclose or suggest transmitting an operand address from a processor to a debug circuit. Section 3 of MotorolaNPL discusses the real-time trace support functionality of the system, but does not say anything about operand addresses or transmitting an operand address from a process to a debug circuit. Similarly, while Section 4 of MotorolaNPL discusses the real-time debug functionality of the system, it does not say anything about operand addresses.

In other portions of the Final Office Action, the Examiner appears to concede that neither Circello nor MotorolaNPL explicitly disclose transmitting an operand address from a processor to a debug circuit, but appears to argue that an instruction address could possibly point to an instruction that includes operand values and that the instruction address could be considered an operand address because the operand values would be stored at the memory location identified by the instruction address. Specifically, the Examiner states, "Examiner has pointed out that an instruction address is the address that points to an operator, operands, or /and data. Therefore, if Applicants try to avoid the teaching by limiting to an address of an operand, or of a byte, or of a bit, it looks like to point to the address of a house, but limits it to the garage."

This position is improper, as it well established that, "[a] claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference." *Vendegaal Bros. v. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987). Further, "[t]he identical invention must be shown in as complete detail as is contained in the...claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Here, the Examiner does not appear to contest the fact that a claim limitation (i.e., the transmission of an operand address from a processor to a debug circuit) is not shown in either of the cited references, nor does the Examiner contend that this limitation is inherent in the prior art (as it is clearly not). If a claim limitation is not shown in the prior art, the claim cannot be anticipated.

D. Neither Circello Nor MotorolaNPL Discloses Or Suggests The Use Of A Single Integrated Circuit

Circello does not disclose that the processor core and debug circuit are included on a single integrated circuit (IC). Rather, Circello merely states that data processor 3 in Figure 1 includes, *inter alia*, a core 9 and a debug module 10 (column 4, lines 7-10). Circello does not disclose whether these elements are implemented on a single IC, on multiple ICs, or in some other different way. Similarly, MotorolaNPL does not state that the processor core and debug circuit are included on a single IC.

E. The Use Of Multiple References In Rejecting Claims 1-64 Under 35 U.S.C. §102(b) Is Improper

Claims 1-64 are rejected under 35 U.S.C. §102(b) as purportedly being unpatentable over Circello and MotorolaNPL. In Appellant's response mailed October 15, 2007, Appellant noted that MPEP §2131.01 indicates that there are three situations when a multiple reference rejection under 35 U.S.C. §102(b) has been held to be proper (i.e., when the additional references are cited to prove the primary reference contains an enabled disclosure, explain the meaning of a term used in the primary reference; or show that a characteristic not disclosed in the reference is inherent). Appellant indicated that it was unclear for which, if any, of these reasons the additional reference is cited and requested that the Examiner explain the purpose for which the Examiner is citing the reference.

In response to this request, the Examiner provided the following explanation on pages 2-3 of the Final Office Action mailed January 10, 2008:

> Furthermore, it should be noted that Circello in the Motorola' Patent and Motorola's NPL is the same person. The references are used in this situation for "enabled disclosure" and for meaning of a term used in the primary reference. It is for showing a characteristic not disclosed in one of a reference is inherent, as addressed in MPEP 2131.01.

Thus, it appears that the Examiner asserts that the secondary reference, MotorolaNPL, is cited for all three purposes stated in MPEP §2131.01 (i.e., to prove the primary reference

contains an enabled disclosure, explain the meaning of a term used in the primary reference; or show that a characteristic not disclosed in the reference is inherent). However, other than simply stating that MotorolaNPL has been cited for these purposes, the Examiner provides no explanation of how MotorolaNPL purportedly achieves any such purposes.

That is, for example, to the extent that the Examiner relies on MotorolaNPL to demonstrate that Circello contains an enabled disclosure, there is no explanation as to what aspects of Circello raise any questions of enablement, or how MotorolaNPL demonstrates that these aspects are enabled. To the extent that the Examiner relies on MotorolaNPL to explain the meaning of the term, the Examiner fails to state what term MotorolaNPL is being relied on to define or what definition MotorolaNPL provides for any such term. To the extent that the Examiner relies on MotorolaNPL to demonstrate that some characteristic is inherent in Circello, the Examiner does not state which characteristic of Circello that is not explicitly disclosed is believed to be inherent, or how MotorolaNPL is believed to demonstrate that any such characteristic is inherent.

Rather, it appears that the Examiner is not relying on MotorolaNPL for any of these purposes, but instead has relied on MotorolaNPL to pick and choose certain claim limitations as purportedly being disclosed by MotorolaNPL. That is, in the explanation of the rejections of the claims, the Examiner cites Circello as purportedly teaching some claim limitations and MotorolaNPL as purporteldy teaching other limitations (*see* Final Office Action of January 10, pages 3-9).

This is improper, as it appears the Examiner has attempted to reject the claims based on some combination of Circello and MotorolaNPL, but has failed to establish what the system resulting from the combination of these two references would like or why one of skill in the art would have combined these two references. Moreover, any rejection relying on the combination of Circello and MotorolaNPL should be made under 35 U.S.C. §103, and not 35 U.S.C. §102.

For this reason the rejection of claims 1-64 under 35 U.S.C. §102(b) is improper and should be reversed.

F. Each Of Claims 1-64 Patentably Distinguishes Over Both Circello and MotorolaNPL

1. Claims 1-20

Claim 1 is directed to a microcomputer comprising: at least one processor; a debug circuit; a system bus coupling the processor and debug circuit; and a communication link coupling the processor and debug circuit, wherein the processor is configured to transmit to the debug circuit through the communication link a plurality of bit values each representing a state of an operation in the processor including at least an operand address that indicates a memory location at which an operand value is stored.

As should be clear from the discussion above, neither Circello nor MotorolaNPL discloses or suggest "a communication link coupling the processor and debug circuit, wherein the processor is configured to transmit to the debug circuit through the communication link a plurality of bit values each representing a state of an operation in the processor **including at least an operand address that indicates a memory location at which an operand value is stored**."

Thus, claim 1 patentably distinguishes over both Circello and MotorolaNPL. Claims 2-20 depend from claim 1 and are patentable for at least the same reasons. Accordingly, it is respectfully requested that the rejections of claims 1-20 be reversed.

2. Claim 21

Claim 21 is directed to a microcomputer implemented on a single integrated circuit. The microcomputer comprises: at least one processor; a debug circuit; a system bus coupling the processor and debug circuit; and a communication link coupling the processor and debug circuit, wherein the processor is configured to transmit to the debug circuit through the communication link a plurality of bit values each representing a state of an operation in the processor including at least one of: an operand address that indicates a memory location at which an operand value is stored; and an operand value. The processor is further configured to transmit to the debug circuit: a program counter value indicating the program counter of the processor at a writeback

stage of a pipeline of the processor; a status indicating that a computer instruction in the writeback stage is a valid computer instruction; a status indicating that the computer instruction in the writeback stage is a first instruction past an executed branch instruction; a status indicating a type of the executed branch instruction; and process identifier information of an executed instruction.

As should be clear from the discussion above, neither Circello nor MotorolaNPL discloses or suggests a microcomputer **implemented on a single integrated circuit** that includes at least one processor; a debug circuit; a system bus coupling the processor and debug circuit; and a communication link coupling the processor and debug circuit.

Thus, claim 21 patentably distinguishes over Circello. Accordingly, it is respectfully requested that the rejection of claim 21 be reversed.

3. Claims 22-41

Claim 22 is directed to a microcomputer comprising: at least one processor; a debug circuit; a system bus coupling the processor and debug circuit; and means for transmitting to the debug circuit a plurality of bit values each representing a state of an operation in the processor including at least an operand address that indicates a memory location at which an operand value is stored.

As should be clear from the discussion above, neither Circello nor MotorolaNPL discloses or suggests means for transmitting to the debug circuit a plurality of bit values each representing a state of an operation in the processor **including at least an operand address that indicates a memory location at which an operand value is stored**.

Thus, claim 22 patentably distinguishes over Circello and MotorolaNPL. Claims 23-41 depend from claim 22 and are patentable for at least the same reasons. Accordingly, it is respectfully requested that the rejections of claims 22-41 be reversed.

4. Claims 42-64

Claim 42 is directed to a method for transferring information between a processor and a debug circuit over a communication link. The method comprises: transmitting to the debug circuit a plurality of bit values each representing a state of an operation in the processor including at least an operand address that indicates a memory location at which an operand value is stored; and transmitting a program counter value indicating the program counter of the processor.

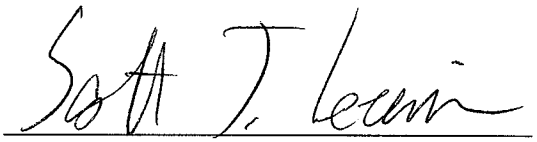
As should be clear from the discussion above, neither Circello nor MotorolaNPL discloses or suggest, transmitting to the debug circuit a plurality of bit values each representing a state of an operation in the processor **including at least an operand address that indicates a memory location at which an operand value is stored.**

Thus, claim 42 patentably distinguishes over Circello. Claims 43-64 depend from claim 42 and are patentable for at least the same reasons. Accordingly, it is respectfully requested that the rejections of claims 42-64 be reversed.

VIII. CONCLUSION

For the foregoing reasons, the rejection of claims 1-64 is improper and should be reversed.

Respectfully submitted,



Scott J. Gerwin, Reg. No. 57,866
James H. Morris, Reg. No. 34,681
WOLF, GREENFIELD & SACKS, P.C.
600 Atlantic Avenue
Boston, MA 02210
Tel: (617) 720-3500
Attorneys for Appellants

Attorney Docket No.: 99-TK-551SS
Dated: June 10, 2008

APPENDIX A – CLAIMS AS PENDING

1. (Previously Presented) A microcomputer comprising:

at least one processor;

a debug circuit;

a system bus coupling the processor and debug circuit; and

a communication link coupling the processor and debug circuit, wherein the processor is configured to transmit to the debug circuit through the communication link a plurality of bit values each representing a state of an operation in the processor including at least an operand address that indicates a memory location at which an operand value is stored.

2. (Original) The microcomputer according to claim 1, wherein at least one of the plurality of bit values represents a state of an operation in the processor including an operand value and operand address.

3. (Original) The microcomputer according to claim 1, wherein the processor is further configured transmit to the debug circuit a program counter value indicating the program counter of the processor.

4. (Original) The microcomputer according to claim 3, wherein the program counter has a value corresponding to a value of the program counter at a writeback stage of a pipeline of the processor.

5. (Original) The microcomputer according to claim 4, wherein the processor is further configured transmit to the debug circuit a status indicating that a computer instruction is in the writeback stage is a valid computer instruction.

6. (Original) The microcomputer according to claim 4, wherein the processor is further configured transmit to the debug circuit a status indicating that the computer instruction in the writeback stage is a first instruction past a branch instruction.

7. (Original) The microcomputer according to claim 6, wherein the processor is further configured transmit to the debug circuit a status indicating a type of an executed branch instruction.

8. (Original) The microcomputer according to claim 7, wherein the debug circuit is configured to transmit a trace packet indicating the type of the executed branch instruction.

9. (Original) The microcomputer according to claim 1, wherein the plurality of bit values representing a pre-execution state of the processor.

10. (Original) The microcomputer according to claim 1, wherein the processor is configured to suppress transmitting the plurality of bit values upon detecting an exception.

11. (Original) The microcomputer according to claim 1, wherein the processor is further configured transmit to the debug circuit address information of an executed instruction.

12. (Original) The microcomputer according to claim 1, wherein the processor is further configured transmit to the debug circuit data information of an executed instruction.

13. (Original) The microcomputer according to claim 1, wherein the processor is further configured transmit to the debug circuit process identifier information of an executed instruction.

14. (Original) The microcomputer according to claim 1, wherein the debug circuit is capable of transmitting processor control signals, including at least one of:

a signal to suspend operation of the processor;

a signal to resume fetching instructions;

a signal to reset the processor;

a signal to indicate that an exception has occurred in the debug unit.

15. (Original) The microcomputer according to claim 1, wherein at least one of the plurality of bit values represents a match state between a match value and a portion of an executed instruction.

16. (Original) The microcomputer according to claim 1, wherein at least one of the plurality of bit values represents a match state between a match value and a memory address accessed by the processor in response to an executed instruction.

17. (Original) The microcomputer according to claim 1, wherein the processor is further configured transmit to the debug circuit a value indicating an increment of the program counter of the processor.

18. (Original) The microcomputer according to claim 1, wherein the processor is further configured transmit to the debug circuit a value indicating a change in process identifier value.

19. (Original) The microcomputer according to claim 3, wherein the debug circuit is adapted to generate trace information including the program counter.

20. (Original) The microcomputer according to claim 1, wherein the microcomputer is implemented on a single integrated circuit.

21. (Previously Presented) A microcomputer implemented on a single integrated circuit, the microcomputer comprising:

at least one processor;

a debug circuit;

a system bus coupling the processor and debug circuit; and

a communication link coupling the processor and debug circuit, wherein the processor is configured to transmit to the debug circuit through the communication link a plurality of bit values each representing a state of an operation in the processor including at least one of:

an operand address that indicates a memory location at which an operand value is stored; and

an operand value;

wherein the processor is further configured transmit to the debug circuit:

a program counter value indicating the program counter of the processor at a writeback stage of a pipeline of the processor;

a status indicating that a computer instruction is in the writeback stage is a valid computer instruction;

a status indicating that the computer instruction in the writeback stage is a first instruction past an executed branch instruction;

a status indicating a type of the executed branch instruction; and

process identifier information of an executed instruction.

22. (Previously Presented) A microcomputer comprising:

at least one processor;

a debug circuit;

a system bus coupling the processor and debug circuit; and

means for transmitting to the debug circuit a plurality of bit values each representing a state of an operation in the processor including at least an operand address that indicates a memory location at which an operand value is stored.

23. (Original) The microcomputer according to claim 22, wherein at least one of the plurality of bit values represents a state of an operation in the processor including an operand value and operand address.

24. (Original) The microcomputer according to claim 22, wherein the microcomputer further comprises means for transmitting to the debug circuit a program counter value indicating the program counter of the processor.

25. (Original) The microcomputer according to claim 24, wherein the program counter has a value corresponding to a value of the program counter at a writeback stage of a pipeline of the processor.

26. (Original) The microcomputer according to claim 25, wherein the processor comprises means for transmitting to the debug circuit a status indicating that a computer instruction is in the writeback stage is a valid computer instruction.

27. (Original) The microcomputer according to claim 25, wherein the processor comprises means for transmitting to the debug circuit a status indicating that the computer instruction in the writeback stage is a first instruction past a branch instruction.

28. (Original) The microcomputer according to claim 27, wherein the processor comprises means for transmitting to the debug circuit a status indicating a type of an executed branch instruction.

29. (Original) The microcomputer according to claim 28, wherein the debug circuit includes means for transmitting a trace packet indicating the type of the executed branch instruction.

30. (Original) The microcomputer according to claim 22, wherein the plurality of bit values representing a pre-execution state of the processor.

31. (Original) The microcomputer according to claim 22, wherein the processor includes means for suppressing a transmission of the plurality of bit values upon detecting an exception.

32. (Original) The microcomputer according to claim 22, wherein the processor further comprises means for transmitting to the debug circuit address information of an executed instruction.

33. (Original) The microcomputer according to claim 22, wherein the processor includes means for transmitting to the debug circuit data information of an executed instruction.

34. (Original) The microcomputer according to claim 22, wherein the processor comprises means for transmitting to the debug circuit process identifier information of an executed instruction.

35. (Original) The microcomputer according to claim 22, wherein the debug circuit comprises means for transmitting processor control signals, including at least one of:

a signal to suspend operation of the processor;

a signal to resume fetching instructions;

a signal to reset the processor;

a signal to indicate that an exception has occurred in the debug unit.

36. (Original) The microcomputer according to claim 22, wherein at least one of the plurality of bit values represents a match state between a match value and a portion of an executed instruction.

37. (Original) The microcomputer according to claim 22, wherein at least one of the plurality of bit values represents a match state between a match value and a memory address accessed by the processor in response to an executed instruction.

38. (Original) The microcomputer according to claim 22, wherein the processor includes means for transmitting to the debug circuit a value indicating an increment of the program counter of the processor.

39. (Original) The microcomputer according to claim 22, wherein the processor is further configured transmit to the debug circuit a value indicating a change in process identifier value.

40. (Original) The microcomputer according to claim 22, wherein the debug circuit includes means for generating trace information including the program counter.

41. (Original) The microcomputer according to claim 22, wherein the microcomputer is implemented on a single integrated circuit.

42. (Previously Presented) A method for transferring information between a processor and a debug circuit over a communication link, the method comprising:

transmitting to the debug circuit a plurality of bit values each representing a state of an operation in the processor including at least an operand address that indicates a memory location at which an operand value is stored; and

transmitting a program counter value indicating the program counter of the processor.

43. (Original) The method according to claim 42, wherein at least one of the plurality of bit values represents a state of an operation in the processor including an operand value.

44. (Original) The method according to claim 43, wherein the program counter has a value corresponding to a value of the program counter at a writeback stage of a pipeline of the processor.

45. (Original) The method according to claim 44, the method further comprises a step of transmitting to the debug circuit a status indicating that a computer instruction is in the writeback stage is a valid computer instruction.

46. (Original) The method according to claim 44, the method further comprising a step of transmitting to the debug circuit a status indicating that the computer instruction in the writeback stage is a first instruction past a branch instruction.

47. (Original) The method according to claim 46, the method further comprising a step of transmitting to the debug circuit a status indicating a type of an executed branch instruction.

48. (Original) The method according to claim 47, the method further comprising a step of transmitting a trace packet indicating the type of the executed branch instruction.

49. (Original) The method according to claim 42, wherein the plurality of bit values representing a pre-execution state of the processor.

50. (Original) The method according to claim 42, the method further comprising a step of suppressing a transmission of the plurality of bit values upon detecting an exception.

51. (Original) The method according to claim 42, the method further comprising a step of transmitting to the debug circuit address information of an executed instruction.

52. (Original) The method according to claim 42, the method further comprising a step of transmitting to the debug circuit data information of an executed instruction.

53. (Original) The method according to claim 42, the method further comprising a step of transmitting to the debug circuit process identifier information of an executed instruction.

54. (Original) The method according to claim 42, the method further comprising a step of transmitting processor control signals, including at least one of:

a signal to suspend operation of the processor;

a signal to resume fetching instructions;

a signal to reset the processor;

a signal to indicate that an exception has occurred in the debug unit.

55. (Original) The method according to claim 42, wherein at least one of the plurality of bit values represents a match state between a match value and a portion of an executed instruction.

56. (Original) The method according to claim 42, wherein at least one of the plurality of bit values represents a match state between a match value and a memory address accessed by the processor in response to an executed instruction.

57. (Original) The method according to claim 42, the method further comprising a step of transmitting to the debug circuit a value indicating an increment of the program counter of the processor.

58. (Original) The method according to claim 42, the method further comprising a step of transmitting a value indicating a change in process identifier value to the debug circuit.

59. (Original) The method according to claim 42, the method further comprising a step of generating trace information including the program counter.

60. (Original) The method according to claim 42, wherein the microcomputer is implemented on a single integrated circuit.

61. (Previously Presented) The microcomputer of claim 1, wherein the plurality of bit values further includes at least an instruction address.

62. (Previously Presented) The microcomputer of claim 21, wherein the plurality of bit values further includes at least an instruction address.

63. (Previously Presented) The microcomputer of claim 22, wherein the plurality of bit values further includes at least an instruction address.

64. (Previously Presented) The method of claim 42, wherein the plurality of bit values further includes at least an instruction address.

APPENDIX B – EVIDENCE

None.

APPENDIX C – RELATED PROCEEDINGS

None.